Manual && Reproduction Report for

Watch and Act: Multi-orientation Open-set Scene Text Recognition via Dynamic Expert Routing

Hardware requirements

GPU:

Testing: NVIDIA GPU with Turing or later structure and more than 4 Gib of vram.

(yes pascals, maxwells or even fermis may work, but you gonna need to know what you are doing)

Training: 24GiB GPU with Turing or later structure

(In theory, 16GiB ones work too but I am not promising that.)

CPU: X86 cpus with AVX2 instruction set

(CPUs without AVX2 may or may not work with different versions of torch, I don't know)

RAM: 16GiB (Testing) 32Gib (Training)

Disk: 500GiB (well just dig out one old hdd from your bucket, or ebay and that's it)

Internet: Yes

Software Requirements

1. Fresh installed Archlinux with these packages:
sudo vim plasma-meta nvidia openssh firefox dolphin konsole wget git less

2. The user name is set to lasercat and it has to be a sudoer (or you may need to go through the code to replace paths if something went south).

3. ***Keep important data off this device!***

I don't want to wipe your data due to one or two failed cd commands followed by mv and/or rm

Trivia: Manul, aka the Pallas' cat, is the oldest cat species still alive on the earth.

Environment setup

1. install the following packages once you boot in.

sudo pacman -Syu pycharm-community-edition plasma-meta python-paramiko python-lmdb python-numba python-opencv python-pillow python-pip python-pyqtgraph python-pytorch-opt-cuda python-regex python-scikit-learn python-scipy python-torchvision-cuda python-tqdm python-xmltodict make gcc cmake unzip python-setuptools

2. make dirs

mkdir ~/cat ~/ssddata ~/hydra_saves ~/ssddata/anchors

3. setup python stuff

mkdir ~/catvenv/; python3 -m venv ${PWD}/catvenv --system-site-packages;

source ~/catvenv/bin/activate; pip install easydict editdistance wandb

cd ~/cat; git clone https://github.com/lancercat/make_envNG.git

cd make_envNG/ ; sh pylcs.sh ;

unzip pytorch_scatter-laser.zip ; cd pytorch_scatter-2.1.2/; python setup.py install

4. unzip data and model

cd ~/cat/wna/ sh unzip.sh ${DOWNLOADPATH} # Note the script is not yet there. Give me some time to test it…

5. clone the code

cd ~/cat; git clone https://github.com/lancercat/wna.git

6. stop and check:

```
(catvenv) [lasercat@TESTMEOW ~]$ ls cat
make_envNG  wna
(catvenv) [lasercat@TESTMEOW ~]$ ls hydra_saves
aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32       pgroute_nd_only-v6S-tiny-AAF
aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_mjst_b32  pgroute_nd_only-v6S-tiny-AAF-01E
aroute_nd_only-v6S-tiny-AAF                                  pgroute_nd_only-v6S-tiny-AAF-01E-run2
aroute_nd_only-v6S-tiny-AAF-01E                              pgroute_nd_only-v6S-tiny-AAF-nedmix
aroute_nd_only-v6S-tiny-AAF-01E-run2                         pgroute_nd_only-v6S-tiny-AAF-nedmix-01E
aroute_nd_only-v6S-tiny-AAF-run2                             pgroute_nd_only-v6S-tiny-AAF-nedmix-01E-run2
aroute_nd_only-v6S-tiny-nedmix-AAF                           pgroute_nd_only-v6S-tiny-AAF-nedmix-run2
aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01                    pgroute_nd_only-v6S-tiny-AAF-run2
aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E                   rule_based-v6S-AAF
aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E-run2              rule_based-v6S-AAF-ohem01E
aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01-run2               rule_based-v6S-AAF-ohem01E-run2
aroute_nd_only-v6S-tiny-nedmix-AAF-run2                      rule_based-v6S-AAF-run2
(catvenv) [lasercat@TESTMEOW ~]$ ls ssddata/
anchors        ctwch          CVPR2016  IC13_1015       mlttrchlat_seen     mlttrkr_hori      rctwtrdb_seen_NG
artdb_seen     ctwdb_seen     dicts     IIIT5k_3000     mlttrchlat_seen_NG  NIPS2014          SVT
artdb_seen_NG  ctwdb_seen_NG  dictsv2   lsvtdb_seen     mlttrjp_hori        pami_ch_fsl_hwdb
athenaNG       CUTE80         IC03_867  lsvtdb_seen_NG  mlttrjp_hv          rctwtrdb_seen
```

Reproducing ablative studies

1. run ablative.py (in a screen session as it takes some time):
python ablative.py 2>&1 | tee all.log
(our log is uploaded to github)

---

2. After the ablative experiments are finished (usually takes like a dozen hours on a GTX 1650), run ablative2table.py to compute mean and standard deviation:

```
/home/lasercat/catvenv/bin/python3.13 /home/lasercat/cat/wna/ablative2table.py
\newcommand{\LPAoBASEoJPNHVoGZSLoLAoAVG}{40.97}
\newcommand{\LPAoBASEoJPNHVoGZSLoLAoSTD}{1.02}
\newcommand{\LPAoWOAROUTEoJPNHVoGZSLoLAoAVG}{42.11}
\newcommand{\LPAoWOAROUTEoJPNHVoGZSLoLAoSTD}{0.51}
\newcommand{\LPAoPGoLoJPNHVoGZSLoLAoAVG}{41.49}
\newcommand{\LPAoPGoLoJPNHVoGZSLoLAoSTD}{1.01}
\newcommand{\LPAoPGoLoSoJPNHVoGZSLoLAoAVG}{43.09}
\newcommand{\LPAoPGoLoSoJPNHVoGZSLoLAoSTD}{0.33}
\newcommand{\LPAoPGoNoJPNHVoGZSLoLAoAVG}{39.66}
\newcommand{\LPAoPGoNoJPNHVoGZSLoLAoSTD}{0.36}
\newcommand{\LPAoPGoNoSoJPNHVoGZSLoLAoAVG}{43.17}
\newcommand{\LPAoPGoNoSoJPNHVoGZSLoLAoSTD}{0.30}
\newcommand{\LPAoARoLoJPNHVoGZSLoLAoAVG}{41.88}
\newcommand{\LPAoARoLoJPNHVoGZSLoLAoSTD}{0.59}
\newcommand{\LPAoARoLoSoJPNHVoGZSLoLAoAVG}{43.90}
\newcommand{\LPAoARoLoSoJPNHVoGZSLoLAoSTD}{0.86}
\newcommand{\LPAoWOOHEMoJPNHVoGZSLoLAoAVG}{42.05}
\newcommand{\LPAoWOOHEMoJPNHVoGZSLoLAoSTD}{0.25}
\newcommand{\LPAoJPNHVoGZSLoLAoAVG}{45.14}
\newcommand{\LPAoJPNHVoGZSLoLAoSTD}{0.35}
\newcommand{\LPAoOHEMoAoJPNHVoGZSLoLAoAVG}{44.53}
\newcommand{\LPAoOHEMoAoJPNHVoGZSLoLAoSTD}{0.31}
------

Process finished with exit code 0
```

---

3. save the output text as a txt file, say reprod.txt
(our log can be found in this github repo as reprod.txt)

---

4. diff them:
diff reprod.txt paperref.txt --color

Expect the results to be a bit different here and there due to float errors between different implementations of operators, which can be caused by different torch versions, gpu structures, vram, etc (yes, torch[1] and CUDNN[2] may choose different backend on different hardware)

[1] https://discuss.pytorch.org/t/different-machines-different-results/100126
[2]https://docs.nvidia.com/deeplearning/cudnn/backend/latest/developer/core-concepts.html

```
(catvenv) [lasercat@TESTMEOW wna]$ diff reprod.txt paperref.txt --color
2c2
< \newcommand{\LPAoBASEoJPNHVoGZSLoLAoSTD}{1.02}
---
> \newcommand{\LPAoBASEoJPNHVoGZSLoLAoSTD}{1.03}
7c7
< \newcommand{\LPAoPGoLoSoJPNHVoGZSLoLAoAVG}{43.09}
---
> \newcommand{\LPAoPGoLoSoJPNHVoGZSLoLAoAVG}{43.07}
9,10c9,10
< \newcommand{\LPAoPGoNoJPNHVoGZSLoLAoAVG}{39.66}
< \newcommand{\LPAoPGoNoJPNHVoGZSLoLAoSTD}{0.36}
---
> \newcommand{\LPAoPGoNoJPNHVoGZSLoLAoAVG}{39.65}
> \newcommand{\LPAoPGoNoJPNHVoGZSLoLAoSTD}{0.35}
13c13
< \newcommand{\LPAoARoLoJPNHVoGZSLoLAoAVG}{41.88}
---
> \newcommand{\LPAoARoLoJPNHVoGZSLoLAoAVG}{41.89}
18,20c18,20
< \newcommand{\LPAoWOOHEMoJPNHVoGZSLoLAoSTD}{0.25}
< \newcommand{\LPAoJPNHVoGZSLoLAoAVG}{45.14}
< \newcommand{\LPAoJPNHVoGZSLoLAoSTD}{0.35}
---
> \newcommand{\LPAoWOOHEMoJPNHVoGZSLoLAoSTD}{0.24}
> \newcommand{\LPAoJPNHVoGZSLoLAoAVG}{45.15}
> \newcommand{\LPAoJPNHVoGZSLoLAoSTD}{0.36}
22a23,24
```

Reproducing Regular Model

run object310-rel/project310_v6SF_stability-re/aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E

```
INFO: starting
/home/lasercat/cat/wna/neko_sdk/neko_framework_NG/modules/concat_mvn_dev.py:67: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True)
  imgt=torch.stack([torch.tensor(i,dtype=this.get_type()) for i in imagelist]).permute(0,3,1,2).contiguous().to(this.mean.data.device)-this.mean;
WARN: Corrupted image for 5171
Date: 2025-06-12 14:36:05.161692 ,TEST: KR-KR-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5170 ,ACR: 0.195357833655706 ,Lenpred_ACR: 0.6969052224371374 ,FPS: 107.1092715124331
WARN: Corrupted image for 5075
Date: 2025-06-12 14:36:52.248106 ,TEST: JPNHV-JPNHV-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5074 ,ACR: 0.45309420575482856 ,Lenpred_ACR: 0.8281434765471029 ,FPS: 107.76280253973862
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 14, 37, 38, 995094), 'TEST': 'JPNHV-JPNHV-OSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7602158828064765, 'R': 0.7063807254434736, 'P': 0.9487908961593172, 'F': 0.8098345727728031}
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 14, 38, 26, 30309), 'TEST': 'JPNHV-JPNHV-GOSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.687875574407918, 'R': 0.6534521158129176, 'P': 0.8634490876986463, 'F': 0.7439148073022313}
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 14, 39, 13, 820), 'TEST': 'JPNHV-JPNHV-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7174177831912302, 'R': 0.802757564151666, 'P': 0.9109083007388092, 'F': 0.8534201954397395}
Date: 2025-06-12 14:39:46.950327 ,TEST: JPN-JPN-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 4009 ,ACR: 0.4784235470192068 ,Lenpred_ACR: 0.8233973559491145 ,FPS: 118.08836023802422
{'Date': datetime.datetime(2025, 6, 12, 14, 40, 20, 974879), 'TEST': 'JPN-JPN-OSR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7983453981385729, 'R': 0.715318869165023, 'P': 0.958590308370044, 'F': 0.8192771084337348}
{'Date': datetime.datetime(2025, 6, 12, 14, 40, 55, 91256), 'TEST': 'JPN-JPN-GOSR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7238437359676695, 'R': 0.6604938271604939, 'P': 0.8923426838514026, 'F': 0.7591099645275717}
{'Date': datetime.datetime(2025, 6, 12, 14, 41, 28, 923666), 'TEST': 'JPN-JPN-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7531380753138075, 'R': 0.8235574630424416, 'P': 0.9421713038734315, 'F': 0.878880407124682}
wandb:
wandb: You can sync this run to the cloud by running:
wandb: wandb sync /home/lasercat/hydra_logs/aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E/watch_and_control/aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E/wandb/offline-run-20250612_143501-93lhcncm
wandb: Find logs at: ../../../../../hydra_logs/aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E/watch_and_control/aroute_nd_only-v6S-tiny-nedmix-AAF-ohem01E/wandb/offline-run-20250612_143501-93lhcncm/logs

Process finished with exit code 0
```

Let’s map them to the table

Split	Registered	Out-of-Set	Name	LA	Recall	Precision	FM
GZSL (Hori.)	*Unique Kanji* Shared Kanji, *Kana*, Latin,	∅	OSOCR-Large [4]	30.83	–	–	–
			OpenCCD-Large [23]	41.31	–	–	–
			OpenSAVR-XL [34]	42.58	–	–	–
			MOoSE-XL* [7]	39.56	–	–	–
			CFOR-XL [37]	44.47	–	–	–
			WnA-S (Ours)*	47.89	–	–	–

Date: 2025-06-12 14:39:46.950327 ,TEST: JPN-JPN-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 4009 ,ACR: 0.4784235470192068 ,Lenpred_ACR: 0.8233973559491145 ,FPS: 118.08836023802422

OSTR (Hori.)	Shared Kanji, *Unique Kanji*	*Kana* Latin	OSOCR-Large [4]	58.57	24.46	93.78	38.80
			OpenSAVR-XL [34]	72.33	72.96	92.62	81.62
			MOoSE-XL* [7]	64.80	80.49	89.12	84.50
			CFOR [37]	71.80	**86.36**	89.52	**87.91**
			WnA-S (Ours)*	75.31	82.30	94.21	87.85

6), 'TEST': 'JPN-JPN-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7531380753138075, 'R': 0.8235574630424416, 'P': 0.9421713038734315, 'F': 0.878880407124682}

GZSL (MO)	Shared Kanji *Unique Kanji*	∅	MOoSE-XL	37.39	–	–	–
			WnA-S (Ours)	45.35	–	–	–

Date: 2025-06-12 14:36:52.248106 ,TEST: JPNHV-JPNHV-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5074 ,ACR: 0.45309420575482856 ,Lenpred_ACR: 0.8281434765471029 ,FPS: 107.76280253973862

OSR (MO)	Shared Kanji Latin	*Unique Kanji* *Kana*	MOoSE-XL	75.56	**74.05**	**95.79**	**83.53**
			WnA-S (Ours)	76.02	70.63	94.87	80.98

'TEST': 'JPNHV-JPNHV-OSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7602158828064765, 'R': 0.7063807254434736, 'P': 0.9487908961593172, 'F': 0.8098345727728031}

GOSR (MO)	Shared Kanji *Unique Kanji*	*Kana*	MOoSE-XL	59.81	63.83	81.89	71.74
			WnA-S (Ours)	68.82	**65.39**	86.35	**74.42**

TEST': 'JPNHV-JPNHV-GOSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.687875574407918, 'R': 0.6534521158129176, 'P': 0.8634490876986463, 'F': 0.7439148073022313}

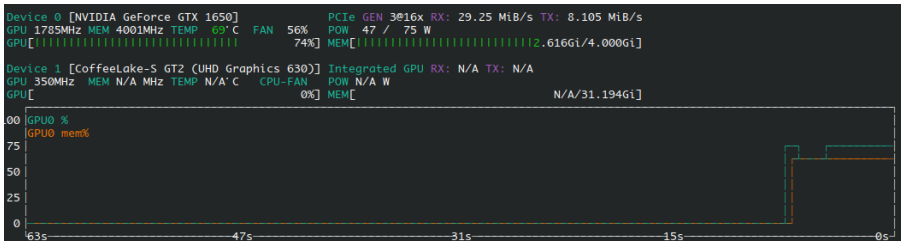
OSTR (MO)	Shared Kanji *Unique Kanji*	Latin *Kana*	MOoSE-XL	61.75	80.01	88.18	83.90
			WnA-S (Ours)	71.78	80.28	91.09	85.34

'TEST': 'JPNHV-JPNHV-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7174177831912302, 'R': 0.802757564151666, 'P': 0.9109083007388092, 'F': 0.8534201954397395}

GZSL (KR)	*Hangul* Latin	∅	CFOR-XL [37]	22.14	–	–	–
			WnA-S (Ours)	19.52	–	–	–

Date: 2025-06-12 14:36:05.161692 ,TEST: KR-KR-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5170 ,ACR: 0.195357833655706 ,Lenpred_ACR: 0.6969052224371374 ,FPS: 107.1092715124331

Note the performance differs a bit on this 1650, but not much (utilization btw)



Reproducing Large Model

Open-set

Run object310-rel/XL/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/osocr_benchall.py

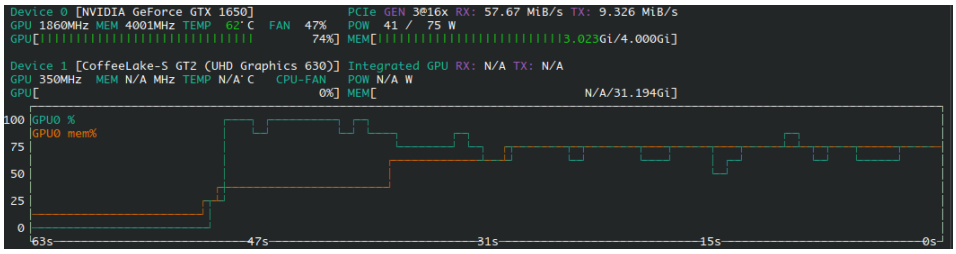
```
/home/lasercat/cat/ena/neko_sdk/neko_framework_NG/modules/concat_evn_dev.py:67: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  img=torch.stack([torch.tensor(i,dtype=this.get_type()) for i in imagelist]).permute(0,3,1,2).contiguous().to(this.mean.data.device)-this.mean;
WARN: Corrupted image for 5171
Date: 2025-06-12 16:06:25.569954 ,TEST: KR-KR-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5170 ,ACR: 0.23965183752417796 ,Lenpred_ACR: 0.7253384912959381 ,FPS: 93.33770637168638
WARN: Corrupted image for 5075
Date: 2025-06-12 16:07:21.041607 ,TEST: JPNHV-JPNHV-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 5074 ,ACR: 0.461568782026015 ,Lenpred_ACR: 0.8303113914071738 ,FPS: 91.47280015247775
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 16, 8, 16, 404025), 'TEST': 'JPNHV-JPNHV-OSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7764070932922128, 'R': 0.7151178183743712, 'P': 0.954416961130742, 'F': 0.8176176782200696}
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 16, 9, 11, 852653), 'TEST': 'JPNHV-JPNHV-GOSR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7122658183103571, 'R': 0.6298440979955456, 'P': 0.8815461346633416, 'F': 0.7347362951415952}
WARN: Corrupted image for 5075
{'Date': datetime.datetime(2025, 6, 12, 16, 10, 7, 506264), 'TEST': 'JPNHV-JPNHV-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 5074.0, 'KACR': 0.7389362565976452, 'R': 0.8050555342780544, 'P': 0.9239560439560439, 'F': 0.8604175194433075}
Date: 2025-06-12 16:10:48.242145 ,TEST: JPN-JPN-GZSL ,Epoch: 0 ,Iter: 0 ,Total: 4009 ,ACR: 0.4804190571214767 ,Lenpred_ACR: 0.823896233474682 ,FPS: 98.4153275062363
{'Date': datetime.datetime(2025, 6, 12, 16, 11, 28, 836917), 'TEST': 'JPN-JPN-OSR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.8097207859358841, 'R': 0.7176199868507561, 'P': 0.9620978684583517, 'F': 0.8220674072679346}
{'Date': datetime.datetime(2025, 6, 12, 16, 12, 9, 855549), 'TEST': 'JPN-JPN-GOSR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7431522227211496, 'R': 0.6234567981234568, 'P': 0.9876797385620915, 'F': 0.7391882980864937}
{'Date': datetime.datetime(2025, 6, 12, 16, 12, 50, 407182), 'TEST': 'JPN-JPN-OSTR', 'Epoch': 0, 'Iter': 0, 'Total': 4009.0, 'KACR': 0.7735355648535565, 'R': 0.8063900810681927, 'P': 0.9489337822671156, 'F': 0.8718741943799947}
wandb:
wandb: You can sync this run to the cloud by running:
wandb: wandb sync /home/lasercat/hydra_logs/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/watch_and_control/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/wandb/offline-run-20250612_160514-78pek56a
wandb: Find logs at: ../../../../../hydra_logs/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/watch_and_control/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/wandb/offline-run-20250612_160514-78pek56a/logs

Process finished with exit code 0
```

Let's map them to the table

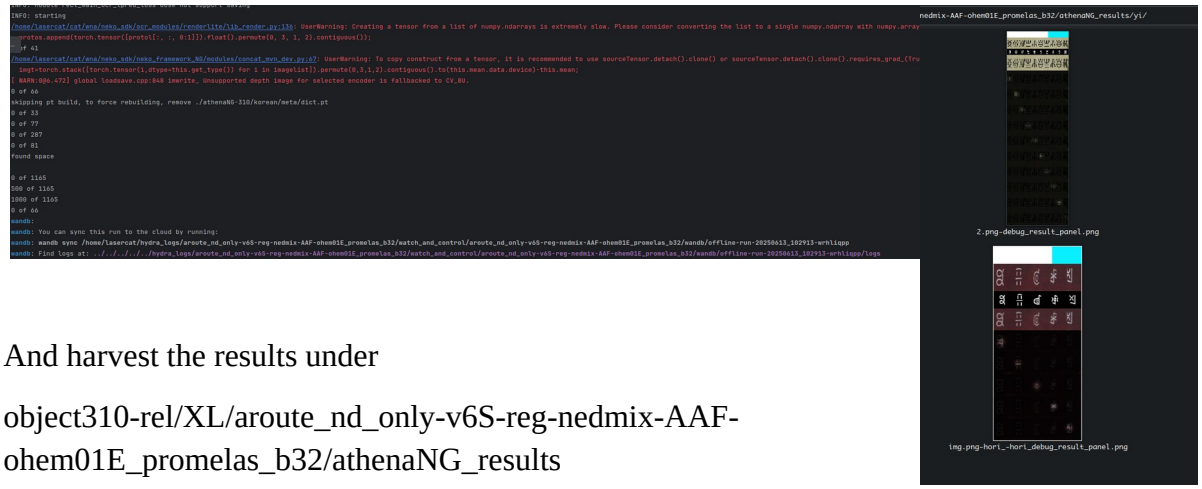
Split	Registered	Out-of-Set	Name	LA	Recall	Precision	FM
GZSL (Hori.)	*Unique Kanji* Shared Kanji, *Kana*, Latin,	∅	OSOCR-Large [4]	30.83	–	–	–
			OpenCCD-Large [23]	41.31	–	–	–
			OpenSAVR-XL [34]	42.58	–	–	–
			MOoSE-XL* [7]	39.56	–	–	–
			CFOR-XL [37]	44.47	–	–	–
			WnA-S-XL (Ours)*	**48.02**	–	–	–
OSTR (Hori.)	Shared Kanji, *Unique Kanji*	*Kana* Latin	OSOCR-Large [4]	58.57	24.46	93.78	38.80
			OpenSAVR-XL [34]	72.33	72.96	92.62	81.62
			MOoSE-XL* [7]	64.80	80.49	89.12	84.50
			CFOR [37]	71.80	**86.36**	89.52	**87.91**
			WnA-S-XL(Ours)*	**77.35**	80.68	**94.89**	87.21
GZSL	Shared Kanji Latin, *Kana*		MOoSE-XL	37.39	–	–	–
			WnA-S-XL(Ours)	**46.14**	–	–	–
OSR	Shared Kanji	*Unique Kanji*	MOoSE-XL	75.56	**74.05**	**95.79**	**83.53**
			WnA-S-XL(Ours)	**77.64**	71.45	95.44	81.72
GOSR	Shared Kanji Latin	*Kana*	MOoSE-XL	59.81	63.83	81.89	71.74
			WnA-S-XL(Ours)	**71.22**	62.89	**88.08**	73.38
OSTR	Shared Kanji	Latin	MOoSE-XL	61.75	80.01	88.18	83.90
			WnA-S-XL(Ours)	**73.89**	**80.50**	**92.39**	**86.04**
GZSL	*Hangul*	∅	CFOR-XL [37]	22.14	–	–	–
			WnA-S-XL (Ours)	**24.00**	–	–	–

Note the performance differs a bit on this 1650, but not much (utilization btw)



Reproducing Fig 1& Inference on your own data

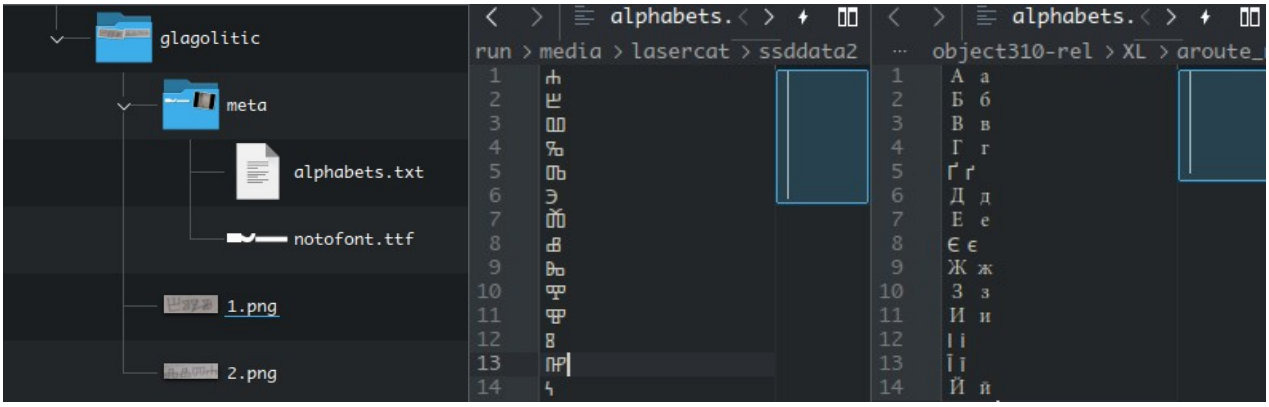
Run object310-rel/XL/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/athena.py



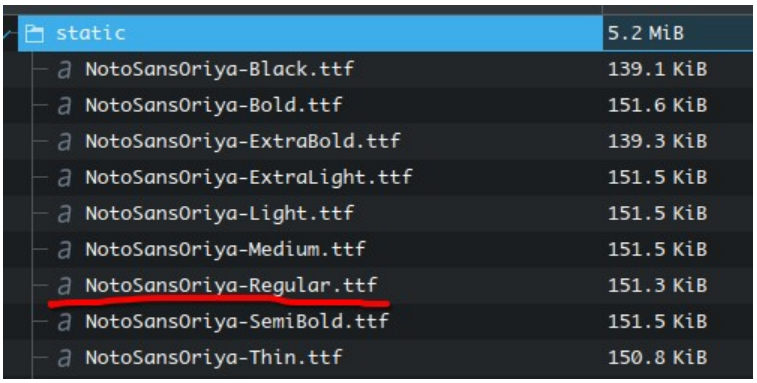
And harvest the results under

object310-rel/XL/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/athenaNG_results

For inferencing your own data, you need to prepare the meta/alphabets.txt, the meta/notofont.ttf, and images.



The meta/alphabets.txt contains all alphabets, each a line. If an alphabet has more than one case, split them with space.



For the notofont.ttf, you search for the font here: https://fonts.google.com/noto Download it, extract the regular variant, rename it to notofont.ttf, and add it to the folder.

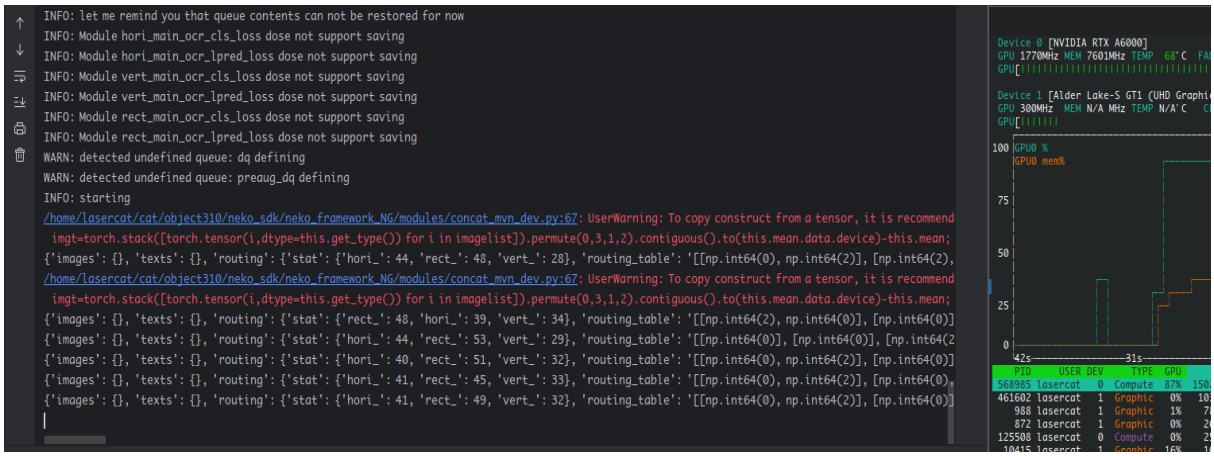
Close-set

Run object310-rel/XL/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_mjst_b32/test.py

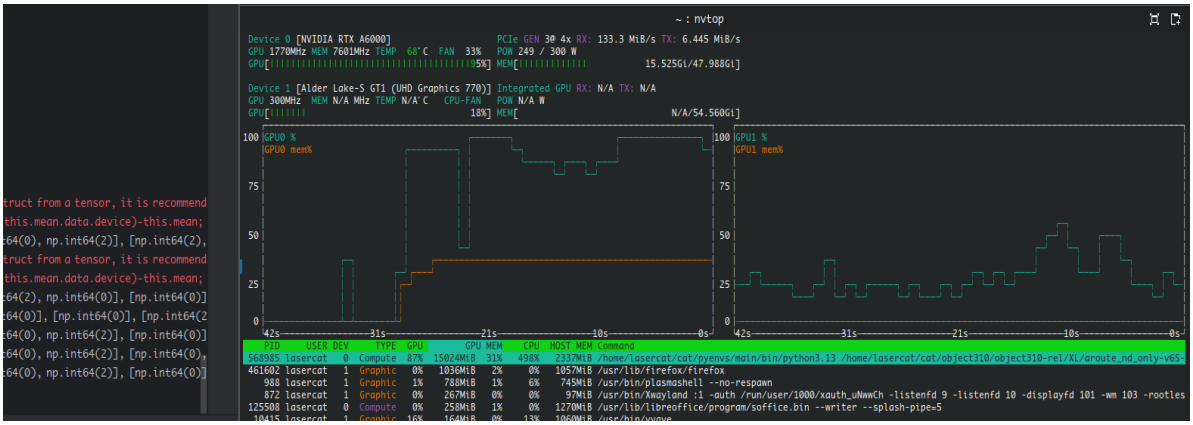
Training from Scratch

Run train.py under the method folder you choose. For example to train the large model:

object310-rel/XL/aroute_nd_only-v6S-reg-nedmix-AAF-ohem01E_promelas_b32/train.py



And watch it cook. Utilization BTW



If you want to train with your own data, please try to follow the instructions from the vanilla osocr-data repo, the lmdbs are fully compatible.

The newly added dicts_v2/lang/vismeta.pt builders can be reverse engineered from the athena scripts. Writing this manual takes too long already so I am not going to write an extensive document on this now… GLHF.

That’s it

If anything is buggy, doesn’t work, or you just want to chat, open an issue or contact us

Chang: lasercat@gmx.us; chang.liu@ltu.se

Elisa: elisa.barney@ltu.se

BTW… Since “manul” is spelled like “manual”, so you will see a manul at the end of each section as an end-of-section icon.

See ya && GLHF